MSc Project - Reflective Essay

Project Title:	Two Players Board Game
Student Name:	Rohit Shindhe
Student Number:	190624325
Supervisor Name:	Matthew Huntbach
Programme of Study:	MSc. Big Data Science

• Analysis of strengths/weaknesses:

ALPHA-BETA pruning is an optimization technique over mini-max algorithm. It has many advantages for using over other methods like native mini-max algorithm.

(i) The main advantage of using alpha-beta pruning is that the unwanted branches of the search tree can be removed/eliminated which saves time.
(ii) The optimization reduces the effective depth of the search tree to slightly more than half that of simple minimax.
(iii) Alpha-Beta pruning can search a tree twice in the same time that mini-max takes to search only once.
(iv) We attain time complexity, with perfect ordering O (b d/2) = O (d).
(v) If b = 40 (as in chess), and the search depth is 12 plies, the ratio between optimal and sorting is a factor of approximately 406 or about 4 billion times.
(vi) Usually iterative deepening is used in conjunction with alpha-beta so that a reasonably good move can be obtained.
(vii) It searches at shallower depths and provide move-ordering hints that can assist to generate cut-offs for higher depth searches which is much earlier than it would otherwise be possible.
(viii) Uses best-first system that possibly makes them more time-proficient, yet normally at an overwhelming expense in space-effectiveness.
(ix) Alpha-beta search can be made significantly quicker by considering only a narrow search window which is known as aspiration search. Aspiration search is generally a guess work based on player's experience.
(x) Adversarial search algorithm is used usually for machine playing of multi-player games like Tic-tac-toe, Chess, etc.
(xi) Adversarial search is used when a quick searching strategy is required.
(xii) The Efficiency will be increased.
(xiii) Time efficiency is quite good. (sabairshad4, 2020)

• Presentation of possibilities for further work:

Longer Games. The dynamic linking device was great at half hour games. The alpha-beta quest tends to be enhanced in hour-length games. While any variation could win, it was likely that the general standard of play was equal. A cut-off on the maximum depth allowed the quest to consider move chains alpha-beta, helped a bit but did not fully eliminate the bug. These findings also suggest changes that could be made to the search process.

Since it was simple to find the bug while playing half hour game, this suggests playing the half hour games is effective. The changes that used move chains search to a depth of 20 ply or more, and in searching this deep there is not much benefit. The evaluation feature cannot construct a better assessment from a limited search. During the hour long games, it is able to produce a better evaluation with more time. So if the move chains measurably won again, the conclusion would be definitive, but as the two variations are rather more equivalent, the argument is that this actually helps to validate

the prior outcomes. It also suggests that there should be a cut-off to be put on the maximum search depth and points out some failings that would occur against stronger searches. (Marckel, n.d.)

The above method shows how the algorithm for wider searches was inferior to the dynamic step sequences. It also indicated that a wider search was needed for it, while a simpler one benefited from the standard alpha-beta search. When the goal is to create a smarter (and more human-like) system, the additional recommendations are suggested. Iterative deepening is also a commonly used heuristic, where for instance 3 ply searches are implemented. It yields the best direction, which is then, at 4 ply, the first direction looked, and fast. If the first best route is checked, then the greatest amount of quest is minimized. Dynamic move sequences check deeply, so why not pursue another complete search, but one that effectively "excludes" the first search route to extend them. The reason will be for the narrow search to skip other valid lines of play, and a user performs this kind of role as well. A player can analyze a pattern of movement to a certain degree, and decide if it was successful or not. Then the player would try a completely new or different track, to see if there was really a better idea. Searching for complex sequences of gestures may be broadened by doing more than one complete search but with the assumption that they are mostly exclusive. For individual movements or smaller move sets, there would need to be some overlap, and so study will need to decide the realistic information of that. In ambiguous games this method is also appealing, as complete searches are returned up to the time limit. This will allow full evaluation of something, when alternative plans could be totally ignored - a more probabilistic and human approach.

Iterative process can also be done in a different way. You can do a different kind of iterative process. The first step, which is the conventional search using move chains, is almost a method of knowledge-extraction that in the second phase is then thought about. Notice that the locations would need to be stored, to identify valid changes between existing sequences of movements, and selection of features will help to define when to cross or what to change. Although this seems to be the reverse of what is done in more traditional searches today, and it uses a version of alpha-beta initial for its tree searches. This also calls for the concept of a "plan" to intelligently combine solution routes, probably more than other search processes. To deal with this, more fundamental information should be available, and statistical scans should also help by automatically eliminating bad or poorly developed plans. (greer, n.d.)

• Work that you would have conducted if you had more time.

While to complete this whole project, it took almost 3 months where I need to study many research papers, IEEE papers, Wikipedia and many other google links including YouTube videos. But as per the requirement I completed this dissertation where a human and a machine can play the chess game with legal moves and with proper time.

If I had more time then I could do more research on how to implement the complete code with different options like:

i. Human to Human: In this multiple players can play with their friends or to practice they have to play themselves from both the ends which makes them to know each and every moves and tactics of movement.
ii. Human to Machine: This one is common game where a player can play with machine using Artificial Intelligence. I have tried and completed this option to my best in my dissertation.
iii. Machine to Machine: This is unique, but I think it should be tried so we can understand how the machines can play and to what extent or depth it moves. As

both players will be machines it will be interesting to know the moves, points and so on.

Implementing with these options I want to make this game online in cloud, where a player (human) can select options like:

- Player want to play with friend or with the machine (AI), before the start of the game in GUI.
- Player want to select White or Black.
- Player wants to 'Flip board' or Disable.

With all the above options I want to implement everything in one game and provide it to general users which could provide more data and efficiency.

• Critical analysis of the relationship between theory and practical work produced

The critical analysis between theory and practical work which I faced is, explanation of the Chess game is easy than to implement it through code.

With the theory, I can explain easily the view of Chess board, coins used, its movements, with all the rules and legal moves. Explanation of complete chess or writing it on paper is easy. Whereas, to make understand for machine it's comparatively difficult.

In practical while coding, I have to take care of which machine and language I are using. While coding I have to implement the board first, so machine can understand it in bit mode. Once board is ready then I have to make sure that all the chess coins are placed on the squares of their respective positions. Once it's set and ready I have to code, so coins can move forward and backwards.

When the coins are able to move, then I have to code separate rule for each coin for their movements as per FIDE. To code this, I faced issues as I wasn't aware of Artificial Intelligence topics.

So, I took some time to understand the concepts from YouTube, online journals, IEEE papers, and e-books. After understanding the concepts like pruning in chess game I was able to code and understand how the game works in efficient way by eliminating the unwanted branches.

• Awareness of Legal, Social Ethical Issues and Sustainability

The new and evolving research area in Human computation is harnessing human intelligence to resolve the computational problems which are beyond the scope of Artificial Intelligence algorithms (AI). With the advancement of the internet we can leverage the human computation abilities to an unprecedented number of people through the web to accomplish complex computations. In today's existing world there are many genres of human computation applications. The games such as ESP game have a purpose specially to target online gamers who can generate useful data while they are playing an enjoyable game. The crowd sourcing market places such as Amazon, Mechanical Turk are human computation systems which coordinate with the workers to perform the tasks which in exchange get the monetary rewards.

To gain the access to some online contents like documents, images, videos users are asked to enter the CAPTCHA, which leverages them to accomplish their tasks. (Human Computation | Synthesis Lectures on Artificial Intelligence and Machine Learning, 2020)

• References

[1] 2019. Alpha-Beta.

[2] sabairshad4, 2020. Alpha Beta. [online] Slideshare.net. Available at: <https://www.slideshare.net/sabairshad4/alpha-beta-131632204> [Accessed 1 September 2020].

[3] Marckel, O., n.d. Alpha-Beta Pruning In Chess Engines.

[4] K. Greer. Tree pruning for new search techniques in computer games. Advances in Artificial Intelligence, 2013:2, 2013

[5] Doi.org. 2020. Human Computation | Synthesis Lectures On Artificial Intelligence And Machine Learning. [online] Available at: <https://doi.org/10.2200/S00371ED1V01Y201107AIM013> [Accessed 2 September 2020].